National Défense
Defence nationale

DESCRIPTION OF A PACKAGE FOR THE ANALYSIS AND DISPLAY OF THE RESULTS OF NUMERICAL SIMULATIONS

by

Marc Dion

47pg

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
TECHNICAL NOTE 93-21

Canada

December 1993
Ottawa

National Defence / Défense nationale

DESCRIPTION OF A PACKAGE FOR THE ANALYSIS AND DISPLAY OF THE RESULTS OF NUMERICAL SIMULATIONS

by

Marc Dion
Countermeasures Section
Electronic Warfare Division

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
TECHNICAL NOTE 93-21

PCN
041LT

December 1993
Ottawa

ABSTRACT

An interactive and versatile program was written for a VAX/VMS system to plot and manipulate the results of some simulation programs. It was first developed to display results from SPICE (a circuit simulation program) but has been modified to accept almost any data in ASCII files. An other version of the program was written for the plotting and manipulation of results during the simulations. This document provides a detailed description of the programs and their commands.

RÉSUMÉ

Un programme a été conçu pour un système VAX/VMS pour afficher et manipuler les résultats de programmes de simulation. Il a été initialement développé pour pouvoir afficher les résultats de SPICE (un programme de simulation de circuits), mais a été ultérieurement modifié pour accepter des données de fichiers ASCII. Une autre version du programme a été écrite pour permettre l'affichage de résultats durant l'exécution de programmes de simulation. Ce rapport donne une description détaillé des programmes et de leurs commandes.

EXECUTIVE SUMMARY

This program (SPLOT) was written at DREO to provide interactive graphics on a VAX/VMS system to display the results of various scientific simulation codes.

SPLOT is available in two forms: as a stand-alone program or as a subroutine. The stand-alone version can display the results of most simulation codes (stored in ASCII files) while the subroutine version easily interacts with user-written codes to display results during the simulation.

SPLOT was first designed to enhance the very limited plotting capability of SPICE. SPICE is a general-purpose circuit simulation program for nonlinear DC and transient analysis, and for linear AC analysis. No changes are required to SPICE and its output can be directly read by SPLOT for plotting.

In addition to the SPICE-format, SPLOT supports ASCII files written in its own format (SPLOT-format) or in free-format.

Some of the features of SPLOT includes the ability to: read multiple data sets from different files, plot multiple curves, select logarithmic axis, scale and zoom plots, manipulate the data (add, move, scale, shift, etc) and edit plot.

TABLE OF CONTENTS

1.0 INTRODUCTION

An interactive and versatile program was developed at the Defence Research Establishment Ottawa (DREO) to manipulate and display in a graphical form the results of simulation programs. It was originally developed to display results from SPICE (a circuit simulation program) but has been expanded to accept data from ASCII files using several different formats.

This program was developed on a VAX/VMS system, using the VAX FORTRAN-77 compiler and using the Tektronix Advance Graphing-II software. This program works well with the Tektronix 401X and 411X series and with the DEC VT2XX and VT3XX series emulating TEK401X serie. It also works well on personal computers with Tektronix emulation software, such as KERMIT or REFLECTION-4. Hardcopy is produced on a DEC LN03 laser printer.

Data can be loaded from an ASCII file, usually generated by a user's application program, conforming to one of the formats defined in Chapter 2. The application program can be written in FORTRAN, C or any other language, as long as it can produce an ASCII file with the required format.

This Technical Note provides a detailed description of the program and its commands.

1.1 GENERAL DESCRIPTION

SPLOT is available in two forms: as a stand-alone program and as a FORTRAN subroutine. The stand-alone version is used to display data previously stored into one or more files. The subroutine version can be linked with a user application and can display data stored in FORTRAN arrays. The subroutine version also has file support, allowing to simultaneously display data from arrays and files. With the exception of the initial loading method and few others exceptions, the two versions are similar and the description in the following sections is applicable to both, except where noted.

SPLOT loads, manipulates and displays data stored into internal arrays (real, single precision). Those arrays are numbered sequentially from 1 to 40

and are noted as A(1) to A(40) throughout this report (A(i..j) denotes the block of arrays i to j). Any array can be referenced and manipulated interactively. A data set consists of a number of arrays (between 2 and 9) of identical size, retrieved from a file or passed as arguments. Each data set is given a unique ID associated with each of the arrays. SPLOT will always load a data set into arrays A(1..9). This data can be moved to other arrays and more data can then be loaded. SPLOT provides a full set of interactive commands to manipulate the arrays and to display up to 16 curves on a single graph.

1.2 RUNNING THE SPLOT PROGRAM

SPLOT is executed by simply entering the "RUN SPLOT" VMS command or by typing "SPLOT" if the user's account has been setup appropriately by the system manager. If a VT2XX or VT3XX series terminal is used, it must be switched to Tektronix emulation mode before the program is started (this can be done automatically if the account is setup properly by the system manager).

Upon execution, the user will be prompted for a file name, from which the initial data set can be loaded.

1.3 CALLING THE SPLOT SUBROUTINE

The subroutine version of SPLOT is almost identical with the program version. The main difference is that the initial data set is loaded from arrays passed as arguments by the caller. The calling syntax is:

```
Call SPLOT2 ( nCurves, Title, Vars, x1 , x2 , .... x8 )
```

where:

nCurves (single precision integer)
The number of data arrays (x1 to x8) passed by the caller

Title (character string)
Title to be displayed on the prompt screen (see below)

Vars (character string)
List of variables to be displayed on the prompt screen (see below)

x1 to x8 (single precision real)
Arrays containing the data points. The first element of each array contains the number of data points, followed by the data itself. Only the number of arrays specified by 'nCurves' need to be passed as arguments. All arrays need not to contain the same number of points, in which case, arrays of different size are treated as separated data sets (and thus given a different ID).

The arrays x1 to x8 can be defined with a different origin to facilitate the programming. For instance, if a user program defines the arrays "REAL x(100), y(100), z(100)", the simplest way to add the size of the array in front of it is to define them as "REAL x(0:100), y(0:100), z(0:100)" and store 100 in x(0), y(0) and z(0).

When linking the application program, the following files must be specified (their exact location can be obtained from the system manager):

SUB-PLOT-V2, PLOT10-UTL/Library, PLOT10/Library

1.4 LOADING DATA

The data can be loaded from different sources (arrays or files), but regardless of the source, SPLOT will attempt to retrieve the following information: a title, a list of variables and a block of data. The title is optional and will be displayed with the command prompt. The list of variables is also optional and, if present, will be displayed also with the command prompt. It identifies the data last loaded into arrays A(1..9). The block of data

contains 2 to 9 columns of 'n' points which will be stored into arrays A(1..9). For instance, if the title and the list of variables are:

```
Title:  SAMPLE TITLE                                  01-Jan-91
List:   t Vin Vout Iin Iout
```

the following screen will appear at the prompt:

```
SAMPLE TITLE                                  01-Jan-91

1 - t

2 - Vin

3 - Vout

4 - Iin

5 - Iout

Command:
```

If neither the title or the list of variables is found, only the "Command: " prompt will be displayed.

2.0 FILE FORMAT DESCRIPTION

SPLOT was originally designed to plot the results of the SPICE circuit simulation program without the need of modifying SPICE itself. It was later modified to accept different file formats, making SPLOT a very versatile tool for displaying the results of various simulation programs. This section describes in detail each of the formats.

2.1 SPICE FORMAT

SPICE is a general-purpose program for simulating passive and active circuits. It can perform nonlinear DC and transient analysis and linear AC analysis. An example of a SPICE input file is shown in Figure 2-1.

The ".PRINT" control statement highlighted in Figure 2-1 must be specified to instruct SPICE to write the desired results into the output file, usually the current or the voltage at one or several nodes. Refer to the SPICE user guide for a complete description of the program.

The results of the simulation are written into an output file, which must have a file name extension ".SPO" (eg. "DEMO.SPO") for SPLOT to recognise its format. SPLOT will automatically locate and load the data generated by the ".PRINT" option. More than one set of data (multiple ".PRINT" command) may be present in the file; they will be loaded sequentially. An extract of a SPICE output file is shown in Figure 2-2.

2.2 SPLOT FORMAT

This format is derived from the SPICE format. It permits the inclusion of titles, lists of variables and multiple sets of data within a single file. This is the most versatile format as it allows one to plot data from multiple data set from one or many files. Files that conform to this format are easily generated by user programs written in C or FORTRAN. Figure 2-3 shows a typical (skeleton) program used to generate a SPLOT file. Typically, a data set in a file is created with three WRITE statements. The first one create a header, the second

```
1******* 1-AUG-88 *******  SPICE 2G.6    3/15/83 *******12:48:49*******

ODEMO 1 -- Simple system: 1st-order & 2nd-order filters in cascade

0****     TRANSIENT ANALYSIS                        TEMPERATURE =   27.000 DEG C

0**********************************************************************************


     TIME        V(1)        V(2)        V(5)
X

  0.000E+00   0.000E+00   0.000E+00   0.000E+00
  1.000E-04   0.000E+00   0.000E+00   0.000E+00
  2.000E-04   0.000E+00   0.000E+00   0.000E+00
  3.000E-04   0.000E+00   0.000E+00   0.000E+00
  4.000E-04   0.000E+00   0.000E+00   0.000E+00
  5.000E-04   0.000E+00   0.000E+00   0.000E+00
  6.000E-04   0.000E+00   0.000E+00   0.000E+00
  7.000E-04   0.000E+00   0.000E+00   0.000E+00
  8.000E-04   0.000E+00   0.000E+00   0.000E+00
  9.000E-04   0.000E+00   0.000E+00   0.000E+00
  1.000E-03   0.000E+00   0.000E+00   0.000E+00
  1.100E-03   0.000E+00   0.000E+00   0.000E+00

     ...         ...         ...         ...

  3.980E-02   0.000E+00   3.832E-18  -2.050E-05
  3.990E-02   0.000E+00   3.142E-18  -1.191E-05
  4.000E-02   0.000E+00   2.551E-18  -2.321E-07
Y
```

Figure 2-2. Sample SPICE output file.

```
DEMO 1 -- Simple system: 1st-order & 2nd-order filters in cascade
*************************************************************************
.OPT     LIST     NODE     LIMPTS=500
VIN      1 0 PULSE(0V 1V 2MS 0MS 0MS 20MS) AC
R1       1 2 50_OHM
C1       2 0 10UF
Ebuf1    3 0 2 0 1
R2       3 4 1.28OHM
L2       4 5 1MH
C2       5 0 25UF
.TRAN    .1MS 40MS
.PRINT   TRAN V(1) V(2) V(5)
.AC      DEC 10 10HZ 1000KHZ
.PRINT   AC VM(2) VM(5)
.END
```

Figure 2-1. Sample SPICE input file.

one included in a DO loop writes the actual data and the last one puts an 'end' marker.

Figure 2-4 shows an extract of a typical data file containing two data sets conforming to the SPLOT format. The first column of every line (the FORTRAN carriage control character) is used as a control character and must be left blank otherwise. The file may contain multiple data sets.

Each data set consists of:

- a form-feed ('1' in column 1), marking the beginning of a new data set.
- an optional title line as 'T' in column 1, followed by a title. For compatibility with older versions of SPLOT, '0' in column 1 is also valid, but discouraged.
- an optional list of variables as 'V' in column 1, followed by the list of variables, each separated with a space.
- some optional comment lines, with a space in column 1.
- 'X' in column 1, marking the beginning of the data area.
- the data area, with a space in column 1, organized as 'n' lines of 'm' columns, which will be loaded into 'm' arrays of 'n' points. Numbers on a line can be separated by spaces, tabs or comma (multiple spaces or tabs are treated as single).
- 'Y' in column 1, marking the end of the data area.

The optional title and variable list are displayed when the user is prompted for commands. They are not part of the graph.

Data points which are greater than 10^{30} are defined as 'not-a-number'. They are stored as data point, but they will not be plotted.

```
      Real            y1, y2, y3, ...

      Open (Unit=2, Name='DEMO.DAT',Type='NEW')

      Write (2,9001)                          ! Write data set header
9001  Format( '1',                            !   Mark beginning of new data set
     +          /,'T Title here'               !   Optional title
     +          /,'V t I1 I2 I3 ...'           !   Optional list of variables
     +          /,'X' )                        !   Mark beginning of data

      Do i= 1, n
        y1 = fct(i)                           ! Generate data here
        y2 = fct1(y1)
        y3 = ...
             ...

        Write (2,9002) y1, y2, y3, ...        ! Output result to file
9002    Format(' ',1p,e10.3, 2x, 8(2x,e10.3) )
      Enddo

      Write (2,9003)                          ! Mark end-of-data
9003  ' .mat('Y')
```

Figure 2-3. Sample program to generate SPLOT format file.

```
1
TSPLOT DEMO. DATA SET #1
Vt V1 V2 V3 V4 V5 V6
X
  0.000E+00    0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+00
  1.000E+00    2.060E+00  2.833E+00  1.477E+00  1.981E+00  2.322E+00  2.585E+00
  2.000E+00    3.308E+00  3.874E+00  2.232E+00  2.814E+00  2.755E+00  2.753E+00
  3.000E+00    4.083E+00  4.458E+00  2.583E+00  2.888E+00  2.570E+00  2.358E+00
  4.000E+00    4.519E+00  4.731E+00  2.648E+00  2.520E+00  2.220E+00  1.896E+00
  5.000E+00    4.793E+00  4.857E+00  2.596E+00  2.287E+00  1.880E+00  1.494E+00
  6.000E+00    4.957E+00  4.915E+00  2.470E+00  1.997E+00  1.538E+00  1.169E+00
  7.000E+00    5.055E+00  4.938E+00  2.309E+00  1.741E+00  1.285E+00  9.115E-01
  8.000E+00    5.112E+00  4.948E+00  2.134E+00  1.508E+00  1.038E+00  7.103E-01
  9.000E+00    5.145E+00  4.948E+00  1.958E+00  1.303E+00  8.507E-01  5.533E-01
  1.000E+01    5.162E+00  4.947E+00  1.788E+00  1.124E+00  6.967E-01  4.309E-01
  1.100E+01    5.171E+00  4.944E+00  1.627E+00  9.685E-01  5.708E-01  3.356E-01
  1.200E+01    5.174E+00  4.940E+00  1.479E+00  8.341E-01  4.672E-01  2.614E-01

     ...          ...        ...        ...        ...        ...        ...

  1.800E+01    5.156E+00  4.911E+00  8.176E-01  3.394E-01  1.407E-01  5.832E-02
  1.900E+01    5.151E+00  4.906E+00  7.400E-01  2.921E-01  1.152E-01  4.542E-02
  2.000E+01    5.148E+00  4.901E+00  6.697E-01  2.514E-01  9.433E-02  3.537E-02
Y
1
TSPLOT DEMO
Vt F1 F2 F3 F4 F5 F6
X
  0.000E+00    0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+00
  1.000E+00   -8.831E-01 -1.053E+00 -8.712E-01 -8.738E-01 -1.014E+00 -1.108E+00
  2.000E+00   -1.418E+00 -1.550E+00 -1.014E+00 -1.165E+00 -1.204E+00 -1.180E+00
  3.000E+00   -1.741E+00 -1.783E+00 -1.165E+00 -1.198E+00 -1.123E+00 -1.010E+00

     ...          ...        ...        ...        ...        ...        ...

  8.000E+00   -2.191E+00 -1.979E+00 -9.698E-01 -8.721E-01 -4.535E-01 -3.044E-01
  9.000E+00   -2.205E+00 -1.980E+00 -8.698E-01 -5.807E-01 -3.716E-01 -2.371E-01
  1.000E+01   -2.212E+00 -1.979E+00 -8.128E-01 -5.008E-01 -3.044E-01 -1.847E-01
Y
```

Figure 2-4. Sample SPLOT format file.

2.3 FREE FORMAT

If SPLOT fails to find in the file a data set as defined in the previous section, it will attempt to interpret the file in free format. It will be assumed that the file contains only data, with no control lines and no control character in column 1. The data must be organized as 'n' lines of 'm' columns. Numbers on a line can be separated by spaces, tabs or comma (multiple spaces or tabs are treated as single). Data points which are greater than 10^{30} are defined as 'not-a-number' and will not be plotted. Data will be read until the end-of-file is reached.

3.0 COMMANDS DESCRIPTION

SPLOT is an interactive program. It provides commands to select curves for plotting, to load data from files, and to perform operations onto the data. It also has the capability to manipulate interactively the plot (zoom-in, add titles, lines or symbols).

At the "Command: " prompt, the user can specify a curve, enter a command or hit "<CR>" to draw the plot. All commands can be abbreviated to four letters. When the plot is completed, the program will either wait until the user hits "<CR>" in normal mode, or automatically enter the cursor mode if it is enabled (see the "CURSOR" command).

Sections 3.2 to 3.5 describe in detail each command available at the "Command: " prompt. When a command requires additional arguments, they must follow the command name, on the same line, separated by a space. The general syntax is:

```
COMMAND a1,a2,...
```

The command name can be abbreviated to four characters. It is followed by a space and then the list of arguments. Arguments may be numerical or alpha-numerical. Alphanumerical inputs consist of all text from the first non-blank character after the command name, to the end of line, including all spaces. All numerical inputs are in free format and separated with commas (","). They may be integer or real numbers. When noted i1, i2 or i3, they represent the index for one of the arrays A(1..40). When some or all of the arguments are optional, they are put in brackets "[ ]", such as:

```
      COMMAND i1[,i2]
or    COMMAND [[x1,]x2]
```

In the first example, the second argument is optional, but the first one must be specified. In the second example, either the first argument or both are optional, i.e. the combinations "x1,x2", "x2", or no arguments, are valid. Note that if the last argument entered is zero, it is considered not present.

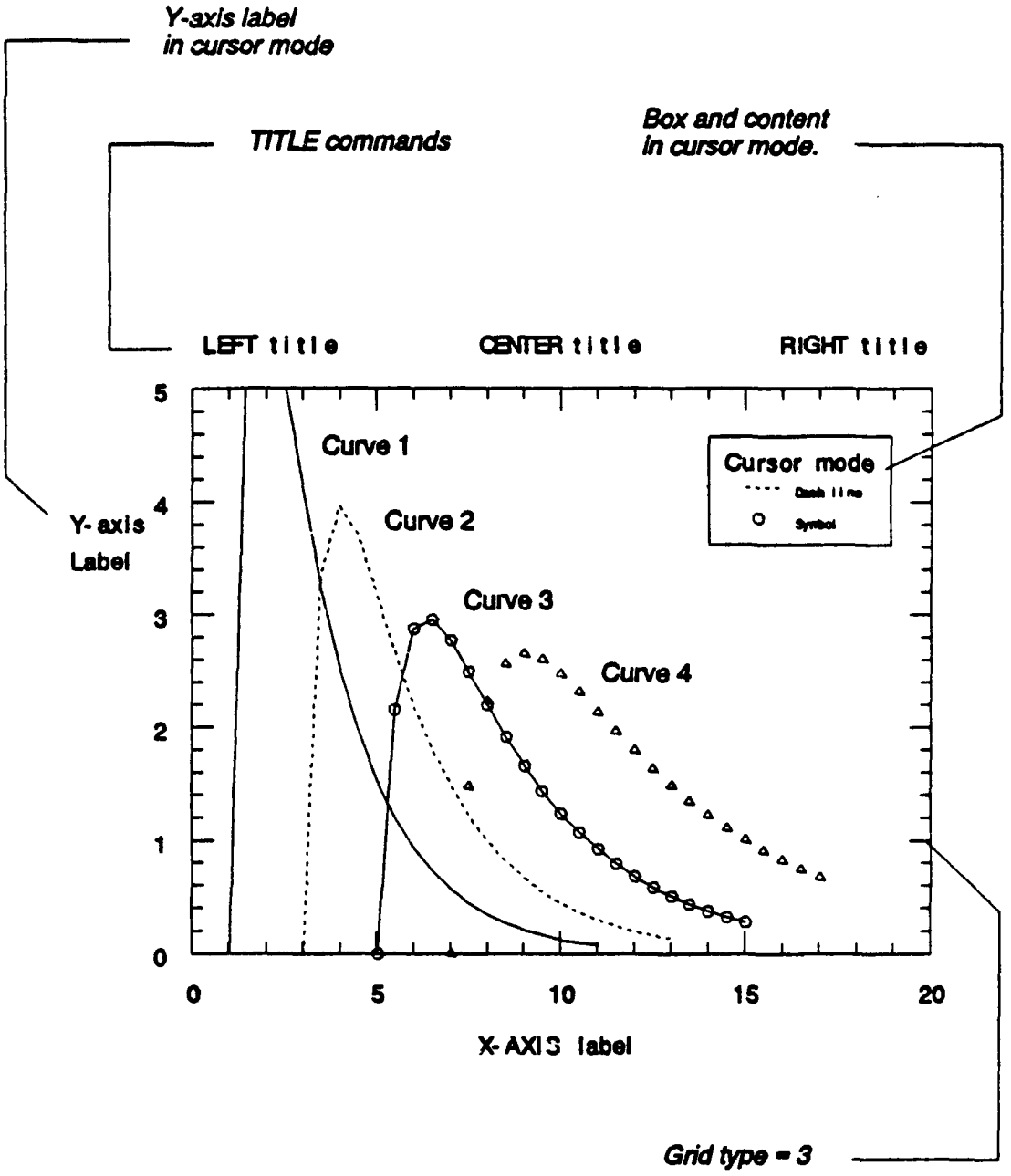

Figure 3-1. Example of a graph produced by SPLOT.

Figure 3-1 shows a typical SPLOT graph of four curves with different line styles or symbols, titles and labels, and some legends added with the cursor mode.

3.1 SPECIFYING CURVES

SPLOT can plot up to 16 curves on the same graph. Data is stored into 40 arrays, A(1) to A(40), which are referred to by specifying their index (1 to 40). Any array can be plotted against any other array having the same ID. One or more curves can be specified by giving the abscissa (x-axis) first, followed by one or more ordinates (y-axis). More than one line can be used to specify more curves or to plot curves from different data set on the same graph. In this example:

$x_1, y_{11}, y_{12}, \cdots y_{1n}$
$x_2, y_{21}, y_{22}, \cdots y_{2m}$
$\cdots \quad \cdots \quad \cdots$

the arrays $A(y_{11})$, $A(y_{12})$, ... $A(y_{1n})$ are plotted against $A(x_1)$ and the arrays $A(y_{21})$, $A(y_{22})$, ... $A(y_{2m})$ are plotted against $A(x_2)$. SPLOT will automatically calculate the minimum and maximum of every curve and setup the graph so they all fit in it (unless some other limits are specified with the "XWINDOW" or "YWINDOW" commands, or with the zoom option of the cursor mode).

Optionally, each array specification may be preceded with a minus sign ("-"):

$[-]x_1, [-]y_{11}, [-]y_{12}, \cdots [-]y_{1n}$
$[-]x_2, [-]y_{21}, [-]y_{22}, \cdots [-]y_{2m}$
$\cdots \quad \cdots \quad \cdots$

If "$-x_1$" is specified, a logarithmic x-axis will be produced. If "$-y_1$" is specified, a logarithmic y-axis will be produced. When an logarithmic axis is specified, the data points of the corresponding arrays which are zero or are negative are ignored. This allows one to obtain for instance a logarithmic plot of a frequency response even if a DC value is included in the data set.

If the "-" sign is specified before any other array, that array will not be considered when computing the minimum and maximum of the axis. Part of the curve may then fall outside the drawing area and will be clipped.

After one or more lines have been entered for specifying curves, an empty line ("<CR>") is used to draw the plot. Upon return to the "Command: " prompt, the curve specifications will remain in effect until new specifications are entered. It is possible to used most of the commands defined in the following sections without affecting the curve specifications. There are however few commands which will clear the specifications (those which loads new data).

3.2 CONTROL COMMANDS

Commands in this group provides general system-level interaction.

3.2.1 <CR>

An empty line initiates the drawing of a plot of all the curves previously specified. When the plot is completed, the terminal will 'beep' and the program will pause until the user hits "<CR>", unless the cursor mode has been enabled (see Section 3.6).

3.2.2 END, EXIT, QUIT or CTRL/Z

For the stand-alone version, this command terminates SPLOT and returns to the DCL prompt. For the subroutine version, control is returned to the caller. Data arrays 10 to 40 and most internal variables are left intact and may be accessed if the subroutine is called again.

3.2.3 PRINT

This command generates a hardcopy of the last plot for a LN03-PLUS laser printer, using the form LN03_TEK (defined by the system manager), on the print

queue LN03$MY_QUEUE (this logical symbol is defined by the user to specify the actual queue). This command should be used immediately after returning to the "Command: " prompt. "PRINT" or "SAVE" can only be called once, unless the plot is drawn again.

3.2.4 SAVE

This command saves the last plot in the file SPLOT.SAV (VMS creates a new file with a new extension every time). The plot is stored in TEK-401X format and can later be sent to a LN03-PLUS printer or redisplayed on a terminal screen. If WordPerfect/VMS is available, this file can also be converted into WP-graphics format (*.WPG files) and be imported into a WordPerfect document or other PC utilities[1]. "PRINT" or "SAVE" can only be called once, unless the plot is drawn again.

3.2.5 SPAWN

This command creates a sub-process and leaves the user at the DCL prompt. Control is returned to SPLOT when the user logs-out ("LOGOUT") the sub-process.

3.3 DATA FILE COMMANDS

Upon entry, an initial data set is always loaded. In the stand-alone version, the user is prompted for a file name. In the sub-program version, the initial data set is passed as arguments. When loading a new data set, the previous content of arrays 1 to 9 is always lost unless the next data set is not found ("end of file detected" condition). The current curve specifications are also lost.

---

[1] The VMS command "WP /Convert=Graphic in_file out_file" is used to convert the file generated by SPLOT into a *.WPG file. The *.WPG file can be copied to a PC and be imported into WordPerfect directly, or can be further edited with other utilities, such as DrawPerfect.

Each data set is given a unique data set ID. Only arrays with the same ID can be plotted against each another.

3.3.1 NEXT

This command reads the next data set from the currently opened file and loads it into arrays 1 to 9. Previous content of arrays A(1..9) is lost.

3.3.2 NEW

This command closes the current file and prompts the user for another file. This first data set found in the file is loaded into arrays 1 to 9. Previous content of A(1..9) is lost.

3.3.3 REWIND

This command 'rewinds' the current file and reads the first data set again into arrays 1 to 9.

3.3.4 BMOVE

This command copies a block of arrays to a different block. A new ID is given to the new block.

Syntax: BMOVE [i1,i2,i3]

This commands copies consecutive arrays A(i1..i2) into a block of arrays starting at A(i3) (arrays A(i3..i3+i2-i1+1)). If no argument is given, the arrays A(1..9) are copied into A(11..19).

3.4 GRAPHIC COMMANDS

Commands in this group change the appearance of the graph. It is possible to change the grid appearance, add titles and labels, plot curves with different line styles and/or symbols. It is also possible to change the range of both the x-axis and y-axis, effectively zooming-in a portion of the graph.

3.4.1 DASH command

This command allows one to specify the style of the line used to draw a curve. Initially, the line style is '0', which correspond to a solid-line. The line style is associated with the array(s) specified, and remains in effect until changed with the "DASH" command, or until the array cleared or superseded. The line style is also affected with the "SCALE" command. The line style is preserved (i.e. copied) when an array is moved into an other one with the "MOVE" command, but it is not preserved with the "BMOVE" command.

Syntax: DASH i1[,i2],icode

where 'i1' and 'i2' represent the index of one array, A(i1), or a range of arrays A(i1..i2), to which a line style 'icode' will be assigned. The Table 1 below lists the possible values of 'icode'. Curve 2 on Figure 3-1 shows the effect of changing the line style. See also the "SYMBOL" command.

The following command is used to specify the line style used by the "SCALE" command:

DASH -1,icode

3.4.2 GRID command

This command sets the appearance of the grid and axis. Figure 3-2 shows the appearance of the various grid types. The default when SPLOT is started is 4. If no argument is given, 4 is assumed.

Value	Line style
≥ 11	Software dash, see below
9	Alternate moves and draws between data points
4	Long dash
3	Medium dash
2	Alternate of short and long dash
1	Short dash
0	Solid line (default)
-1	No line
-2	Vertical bar
-3	Horizontal bar
-4	Point plot
-5	Solid line (same as 0)

Software dashes are obtained by concatenating integers describing the line segment and visibility by alternating visible and invisible segments.

1	5 raster units visible
2	5 raster units invisible
3	10 raster units visible
4	10 raster units invisible
5	25 raster units visible
6	25 raster units invisible
7	50 raster units visible
8	50 raster units invisible

Table 1. Values for specifying line types.

Syntax: GRID [i1]

where 'i1' is a code between 0 and 4:

0: Only the bottom and left axis are drawn, with tic marks on the outside.

1: Only the bottom and left axis are drawn, with tic marks on the inside.

2: Box drawn around graph, with tic marks on the bottom and left, on the inside.

3: As '2', but the axis x=0 and y=0 are also drawn if they fall within the graph.

4: Box and grid drawn, with tic marks on the bottom and left.

3.4.3 SYMBOL command

This command allows one to specify a symbol to be drawn at every data points. Initially, no symbols are drawn and only the connecting lines between data points are drawn. The symbol value is associated with the array(s) specified, and remains in effect until changed with the "SYMBOL" command, or until the array cleared or superseded. The symbol value is preserved (i.e. copied) when an array is moved into an other one with the "MOVE" command, but it is not preserved with the "BMOVE" command.

Syntax: SYMBOL i1[,i2],icode

where 'i1' and 'i2' represent the index of one array, A(i1), or a range of arrays A(i1..i2), to which a symbol value 'icode' will be assigned. Table 2 below lists the possible values of 'icode'. Symbols can be plotted with or without a connecting line. Curves 3 and 4 on Figure 3-1 shows the effect of using a symbol with and without a line. See also the "DASH" command.

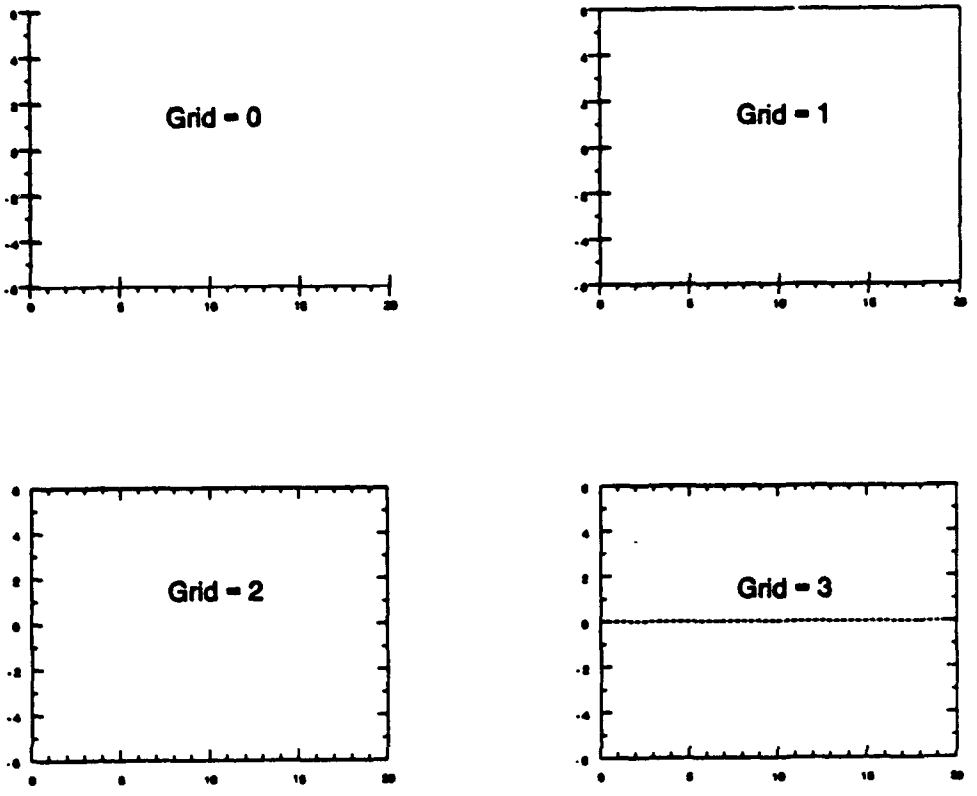

Figure 3-2. Appearance of the grid and axis.

Value	Symbol	
-1 or 0		None
1	o	Circle
2	x	"X"
3	▲	Triangle
4	□	Box
5	*	Star
6	◇	Diamond
7	|	Vertical bar
8	+	Cross
9	↑	Up arrow
10	↓	Down arrow
11	▼	Reverse triangle
33 ≤ value ≤ 127		Corresponding ASCII character centered on the point.

Table 2. Values for specifying symbols.

It is also possible to specify the size of all symbols (1 being the default size) with the command:

SYMBOL -1,size

where 'size' is a relative size compared with the default size.

3.4.4 TITC command

This command displays a title centered on the first line. It can be used in conjunction with the "TITL" and "TITR" commands.

Syntax: TITC [text]

where 'text' is a line of up to 60 characters, which can include spaces. This field is initially empty. It is also cleared with the "NEW" command with the stand-alone version only.

3.4.5 TITL command

This command displays a title on the left of the first line. It can be used in conjunction with the "TITC" and "TITR" commands.

Syntax: TITL [text]

where 'text' is a line of up to 60 characters, which can include spaces. This field is initially empty for the subroutine version and initially set to the initial data file name for the program version. The "NEW" command also sets this field with the file name for both versions.

3.4.6 TITR command

This command displays a title on the right of the first line. It can be used in conjunction with the "TITC" and "TITL" commands.

Syntax: TITR [text]

where 'text' is a line of up to 60 characters, which can include spaces. This field is initially set to the current date (on first entry only; subsequent entry of the subroutine version leaves this field unchanged). The "NEW" command of the stand-alone version also sets this field with the current date.

3.4.7 XLABEL command

This command displays a label centered below the x-axis.

Syntax: XLABEL [text]

where 'text' is a line of up to 60 characters, which can include spaces.

3.4.8 XSCALE command

This command scales the x-axis (for all curves). For example, a scaling factor of 10^9 will draw the x-axis in nanosecond.

Syntax: XSCALE [s]

where 's' is the scaling factor (real number). With no argument, the scaling factor is cleared.

3.4.9 XWINDOW command

This command zooms-in a portion of the x-axis. It allows one to specify the range of the x-axis. It supersedes the autoranging function defined in Section 3.1.

Syntax: XWINDOW [[x_{min},]x_{max}]

where 'x_{min}' and 'x_{max}' specify the range (real numbers). If 'x_{min}' is not specified, the range will be 0. to 'x_{max}'. If no argument is given, the zoom function is cancelled and autoranging resumes.

3.4.10 YSCALE command

This command scales the y-axis (for all curves). For example, a scaling factor of 0.001 will draw the y-axis in millivolts.

Syntax: YSCALE [s]

where 's' is the scaling factor (real number). With no argument, the scaling factor is cleared.

3.4.11 YWINDOW command

This command zooms-in a portion of the y-axis. It allows one to specify the range of the y-axis. It supersedes the autoranging function defined in Section 3.1.

Syntax: YWINDOW [[y_{min},]y_{max}]

where 'y_{min}' and 'y_{max}' specify the range (real numbers). If 'y_{min}' is not specified, the range will be 0. to 'y_{max}'. If no argument is given, the zoom function is cancelled and autoranging resumes.

3.5 DATA MANIPULATION COMMANDS

Commands in this group alter the content of one or more arrays by performing operations onto them.

3.5.1 ABS command

This command computes the absolute value of an array, A(i1), and stores the result back into it.

Syntax: ABS i1

3.5.2 ADD, SUB, MUL and DIV commands

These commands add, subtract, multiply or divide two arrays, A(i1) op A(i2), and store the result back into A(i2).

Syntax: ADD i1,i2

3.5.3 ADDC, SUBC, MULC and DIVC commands

These commands add, subtract, multiply or divide an array, A(i1), with a constant, and store the result back into A(i1).

Syntax: ADDC i1,c

3.5.4 COMPRESS command

This command operates on all curves (i.e. a complete data set) having the same ID as the specified curve, A(i1). It decimates (i.e. compress) all arrays by sub-indexing, i.e. for arrays on 'n' points, stored in A(i;1..n), only the data points A(i;k), for k from 'start' to 'stop', incremented by 'step' are kept (i specifies all curves with the same ID). See also the "XCOMPRESS" command.

Syntax: COMPRESS i1,step or

COMPRESS i1,start,stop[,step]

If only 'step' is specified, 'start=1' and 'stop=n' are assumed. If only 'start' and 'stop' are specified, 'step=1' is assumed.

3.5.5 DB command

This command computes the decibel value ($10 \cdot LOG_{10}x$) of an array, A(i1), and stores the result back into it. Data points less or equal than zero are assigned the 'not-a-number' value and will not be plotted.

Syntax: DB i1

3.5.6 INV command

This command computes the multiplicative inverse (1/x) of an array, A(i1), and stores the result back into it. Data points equal to zero are assigned the 'not-a-number' value and will not be plotted.

Syntax: INV i1

3.5.7 MOVE or COPY commands

The "MOVE" command ("COPY" is a synonymous) copies the content of an array, A(i1), into another array, A(i2). The content of A(i2) is lost. The line style and the symbol code are also copied along. The new array, A(i2), retains the same ID as A(i1). See also the "BMOVE" command.

Syntax: MOVE i1,i2

3.5.8 NEG command

This command computes the negates (-x) an array, A(i1), and stores the result back into it.

Syntax: NEG i1

3.5.9 PHASE command

This command assumes that the content of the specified array, A(i1), represent a phase curve in degrees. Often, when a phase curve is generated numerically, it results in a curve that wraps-around introducing steps of 360° due to the inherent characteristics of the inverse trigonometric functions (they may return an angle in the range $-180 < \theta \leq +180$ or $-270 < \theta \leq +90$ for instance). In some other algorithms, the phase step may be different than 360°. To cover both cases, two algorithm to unwrap the phase were implemented. The first one assumes that the phase jumps by ±360° and will add or subtract a 360° offset when

needed to unwrap the curve. The second algorithm will locate jumps larger than 25° and extrapolate the curve from both side to compute the offset. The second algorithm will fail if not enough data points are given when the phase changes rapidly.

Syntax: PHASE i1

The following command is use to specify which algorithm to use:

PHASE -1,n

where 'n=1' specifies the first algorithm (default) and 'n=2' specifies the second one.

3.5.10 ROT command

This command (rotate) performs a rotation of the data of an array, A(i1), by 'n' points to the right with the end wrapping to the beginning if positive, or a rotation to the left with the beginning wrapping to ths end if negative. See also the "SHIFT" command.

Syntax: ROT i1,n

3.5.11 SCALE command

This command multiplies an array, A(i1) with a factor such as its peak value (positive or negative) matches the peak value of an other array, A(i2). The resulting array is written back into A(i1), and A(i2) is unchanged. By default, the line style of A(i1) is set to dash line (see "DASH" command).

Syntax: SCALE i1[,i2]

If 'i2' is not specified, the scaling factor computed the last time the "SCALE" command was called is used.

3.5.12 SHIFT command

This command performs a shift of the data of an array, A(i1), by 'n' points to the right if positive or to the left if negative. When shifting to the right (left), the 'n' rightmost (leftmost) points are discarded and the 'n' leftmost (rightmost) points are set to 'not-a-number'. See also the "ROT" command.

Syntax: SHIFT i1,n

3.5.13 XCOMPRESS command

This command operates on all curves (i.e. a complete data set) having the same ID as the specified curve, A(i1). It decimates (i.e. compress) all arrays by sub-indexing, i.e. for arrays on 'n' points, stored in A(i;1..n), only the data points (in all arrays with the same ID) for which A(i;k) is between 'xmin' and 'xmax' are kept. See also the "COMPRESS" command.

Syntax: XCOMPRESS i1,[xmin,]xmax

where 'xmin' and 'xmax' are the limits against which the array A(i1) will be checked. If 'xmin' is not specified, it is assumed to be 0.

3.6 CURSOR MODE COMMANDS

When a plot is completed, the program (in normal mode) will signal the user ('beep') and wait for a "<CR>". In addition, SPLOT has the capability to enter what is called the cursor mode, in which it is possible to interactively add elements to the graph, such as additional titles and legends.

Cursor mode can be entered by entering "C" and then "<CR>" after the plot is completed. A cross-hair cursor will be displayed which can be moved around by using the thumbwheels or joydisk on TEK401X and TEK411X series terminals, or by using the array keys on DEC VT2XX and VT3XX terminals and on PC with emulation software. Cursor mode can also be entered automatically upon completion of the plot with the "CURSOR" command.

All commands in cursor mode are one letter long (no "<CR>") possibly followed with an other letter or numerical entry. To return to the "Command: " prompt, enter a single space, "<SP>", without a "<CR>".

All additions made to the plot are shown on screen and can be either saved in a file or sent to the printer. These additions are in effect for the current plot only and they will not appear on subsequent plots.

3.6.1 CURSOR command

Upon execution of this command, the cursor mode will be entered automatically upon completion of the plot.

3.6.2 <SP>, Exit cursor mode

In cursor mode, the space, "<SP>", is used instead of "<CR>" to return to the "Command: " prompt.

3.6.3 <C> command

In normal mode, when waiting for a "<CR>" to return to the "Command: " prompt, typing "C" and "<CR>" will enter the cursor mode. Cursor mode will not be entered automatically after subsequent plots.

3.6.4 <D> command

This command draws a dash line from the last stored cursor position (with the last "<D>", "<L>" or "<M>" commands) to the current cursor position.

3.6.5 <D> command

This command draws a horizontal dash line across the plot passing through the current cursor position.

3.6.6 <L> command

This command draws a solid line from the last stored cursor position (with the last "<D>", "<L>" or "<M>" commands) to the current cursor position. This command is particulary useful to have text point an area on the graph.

3.6.7 <M> command

This command stores the current cursor position for further use with "<D>" and "<L>" commands.

3.6.8 <P> command

Send this plot to the printer and return to the "Command: " prompt. See also the "PRINT" command.

3.6.9 <R> command

This command resets the x-axis and y-axis limits (see the "XWINDOW", "YWINDOW" and "<Z>" commands) and redraws the plot using autoranging.

3.6.10 <S> command

This command draws a symbol at the current cursor position. After "S" is typed, the program waits for the user to enter the symbol value, followed by a "<CR>"; the user entry is not echoed on screen. See the "SYMBOL" command for

a description of the possible values. The size of the symbol will be the same as the size specified with the "SYMBOL" command.

3.6.11 <V> command

This command draws a vertical dash line across the plot passing through the current cursor position.

3.6.12 <Z> command

This command is used to define an area of the graph for zoom-in. Move the cursor anywhere within the graph area and enter "Z". A vertical and horizontal line will be draw on screen to make a box along with the crosshair cursor. Position the cursor again and enter "<SP>". The corners of the box will be used to change the limits of both the x-axis and y-axis, as with the "XWINDOW" and "YWINDOW" commands. If the cursor is moved horizontally only, the limits for the y-axis will not be changed; if it is moved vertically only, the limits for the x-axis will not be changed. If "C" in entered instead of "<SP>", the operation is cancelled.

3.6.13 <'> command

This command is used to display a line of text starting at the current cursor position. "<CR>" is used to terminate entry of text. Small characters are used.

3.6.14 <"> command

This command is used to display a line of text starting at the current cursor position. "<CR>" is used to terminate entry of text. Large characters are used.

4.0 CONCLUSION

Most simulation codes available on mainframe computers lack the data analysis and display capabilities. Many of these codes, such as SPICE, NEC, EFIE, etc., produce an output file in a formatted form which, when printed, can be a few inches thick. Furthermore, it is frequent for researchers to design their own numerical simulation tools to meet their very special requirements and addition of display capability requires considerable effort and thus, is not generally done.

The programs described in this report were designed to add analysis and display capability to almost any simulation code. They can also be incorporated into other codes to provide the same capabilities dynamically, ie. while the code is executing. The data is obtained from the outputs of one or more simulation codes, allowing the comparison of results from different runs or different codes running the same problem. The data can be manipulated to extract and display the desired quantities.

These programs have become our primary tool for analyzing computer generated data and have considerably increased the usefulness of many simulation codes as they greatly reduced the time required to analyze the results.

APPENDIX A

COMMAND SUMMARY

CONTROL COMMANDS

END EXIT QUIT or CTRL/Z	Terminate SPLOT
PRINT	Send the last plot to a LN03+ printer
SAVE	Save the last plot in a file
SPAWN	Suspend execution & spawn a subprocess

DATA FILE COMMANDS

NEXT	Read the next data set from file
NEW	Open a new file and read the first data set
REWIND	Read the first data set again
BMOVE [i1,i2,i3]	Move a block of arrays A(i1..i2) in A(i3...)

GRAPHICS COMMANDS

DASH i1[,i2],icode	Specify line type
GRID icode	Specify grid and axis type
SYMBOL i1[,i2],icode	Specify symbol code (i1=-1 to specify size)
TITC [text]	Specify centered title
TITL [text]	Specify left title
TITR [text]	Specify right title
XLABEL [text]	Specify x-axis label
XSCALE s	Specify scale factor for x-axis
XWINDOW [xmin[,xmax]]	Specify x-axis range
YSCALE s	Specify scale factor for y-axis
YWINDOW [ymin[,ymax]]	Specify y-axis range

DATA MANIPULATION COMMANDS

ABS i1	(i1) ← |(i1)|
ADD i1,i2	(i2) ← (i1) + (i2)
ADDC i1,c	(i1) ← (i1) + c
COMPRESS i1,[start,stop,][step]	Compress data set by index
DB i1	(i1) ← 20·LOG((i1))
DIV i1,i2	(i2) ← (i1) / (i2)
DIVC i1,c	(i1) ← (i1) / c
INV i1	(i1) ← 1 / (i1)
MOVE i1,i2	(i2) ← (i1)
MUL i1,i2	(i2) ← (i1) × (i2)
MULC i1,c	(i1) ← (i1) × c
NEG i1	(i1) ← -(i1)
PHASE i1	Adjust phase (i1=-1 to specify algorithm)
ROT i1,n	Rotate data (positive=right)
SCALE i1[,i2]	Scale (i1) to fit (i2)
SHIFT	Shift data (positive=right)
SUB i1,i2	(i2) ← (i1) - (i2)
SUBC i1,c	(i1) ← (i1) - c
XCOMPRESS i1,[xmin,]xmax	Compress data set by value (xmin≤(i1)≤xmax)

CURSOR MODE COMMANDS

CURSOR	Enable automatic cursor mode
C	Enter cursor mode
D	Draw a dash line
H	Draw a horizontal dash line
L	Draw a line
M	Set cursor position for 'D' & 'L'
P	Print graphic
R	Reset x & y limits
S	Draw a symbol
V	Draw a vertical dash line
Z	Zoom-in
'	Write text on screen (small characters)
"	Write test on screen (large characters)
<SP>	Exit cursor mode

REFERENCES

[1] "*SPICE Version 2G User's Guide*", University of California

[2] "*PLOT-10 Terminal Control System*", Tektronix, document 062-1474-00, 1974

[3] "*PLOT-10 Advance Graphing II, User's Manual*", Tektronix, document 062-1530-00, 1975

[4] "*VAX FORTRAN User Manual*" and "*VAX FORTRAN Language Reference Manual*", Digital Equipment Corporation, June 1988

[5] "*WordPerfect V5.0 for VAX/VMS Systems*", WordPerfect Corporation, 1990

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM
(highest classification of Title, Abstract, Keywords)

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Defence Research Establishment Ottawa
Ottawa, Ontario
K1A 0Z4

2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

Description of a Package for the Analysis and Display of the Results of Numerical Simulations (U)

4. AUTHORS (Last name, first name, middle initial)

Dion, Marc

5. DATE OF PUBLICATION (month and year of publication of document)

December 1993

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

43

6b. NO. OF REFS (total cited in document)

5

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.

DREO Technical Note 93-21

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.

DREO

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

Project 041LT

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). however, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

Unlimited Announcement

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

RA.W (24 Nov 93)

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM

13. **ABSTRACT** (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

An interactive and versatile program was written for a VAX/VMS system to plot and manipulate the results of some simulation programs. It was first developed to display results from SPICE (a circuit simulation program) but has been modified to accept almost any data in ASCII files. An other version of the program was written for the plotting and manipulation of results during the simulations. This document provides a detailed description of the programs and their commands.

RÉSUMÉ

Un programme a été conçu pour un système VAX/VMS pour afficher et manipuler les résultats de programmes de simulation. Il a été initialement développé pour pouvoir afficher les résultats de SPICE (un programme de simulation de circuits), mais a été ultérieurement modifié pour accepter des données de fichiers ASCII. Une autre version du programme a été écrite pour permettre l'affichage de résultats durant l'exécution de programmes de simulation. Ce rapport donne une description détaillé des programmes et de leurs commandes.

14. **KEYWORDS, DESCRIPTORS or IDENTIFIERS** (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Plotting Program for VAX/VMS
SPICE (plotting program for)
VAX/VMS plotting program

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM